

Modelling and Optimal Control of a Docking Maneuver with an Uncontrolled Satellite

Johannes Michael * Kurt Chudej ** Jürgen Pannek ***

* *Universität der Bundeswehr, Fakultät für Luft- und Raumfahrttechnik, Institut für Mathematik und Rechneranwendung, München, Germany (e-mail: johannes.michael@unibw.de)*

** *Universität Bayreuth, Fakultät für Mathematik, Physik und Informatik, Lehrstuhl für Ingenieurmathematik, Bayreuth, Germany (e-mail: kurt.chudej@uni-bayreuth.de)*

*** *Universität der Bundeswehr, Fakultät für Luft- und Raumfahrttechnik, Institut für Mathematik und Rechneranwendung, München, Germany (e-mail: juergen.pannek@unibw.de)*

Abstract: Capturing disused satellites in orbit and their controlled reentry is the aim of the DEOS space mission. Satellites that ran out of fuel or got damaged pose a threat to working projects in orbit. Additionally, the reentry of such objects endangers the population as the place of impact cannot be controlled anymore. This paper demonstrates the modelling of a rendezvous szenario between a controlled service satellite and an uncontrolled target. The situation is modelled via first order ordinary differential equations where a stable target is considered. In order to prevent a collision of the two spacecrafts and to ensure both satellites are docked at the end of the maneuver, additional state constraints, box constraints for the control and a time dependent rendezvous condition for the final time are added. The problem is formulated as an optimal control problem with Bolza type cost functional and solved using a full discretization approach in AMPL/IpOpt. Last, simulation results for capturing a tumbling satellite are given.

Keywords: optimal control, satellite control, modelling, differential equations

1. INTRODUCTION

According to NASA's *Orbital Debris Program*, see NASA Orbital Space Debris Program (2011), approximately 19.000 known objects with diameters larger than ten centimeters and estimated 500.000 objects between one and ten centimeters are currently in orbit. Due to their average speed of 10 km/s even such small objects may cause serious damage to working satellites or the international space station. Additionally, with each collision the number of debris increases rapidly as the cloud of around 600 fractions of the 'Cosmos 2251' and 'Iridium' crash in 2009, see Pickup (2009).

Aside from orbital collisions, an uncontrolled reentry into the atmosphere poses a severe threat. As the entrance corridor is highly difficult to precalculate, with every crash pieces of beryllium or titan may reach the earth's surface and hit populated areas. In October 2011 the german satellite 'Rosat' crashed uncontrolled and even a few hours prior to impact experts were unsure where fractions will strike the earth, see Klinkrad (2011).

The planned german technology demonstration mission DEOS (**D**eutsche **O**rbitale **S**ervicing **M**ission) takes the next step towards capturing and controlling disused satellites. Considered as a feasibility study, DEOS consists of two spacecrafts, one acting as uncontrollable target and one as servicer. The aims of the mission are to demonstrate

docking and berthing with a noncooperating target and the controlled reentry into atmosphere in coupled condition, see Sellmaier et al. (2010).

This paper presents a model to calculate the optimal control schemes for a DEOS capturing mission. In particular, the equations for a rendezvous maneuver to an uncontrollable target are derived and solved for one example with a direct approach. The presented results show the feasibility of such a mission. Additionally, reference trajectories and control schemes for different situations are obtained which are extendable to the case of a tumbling target.

The following section deals with the derivation of the optimal control problem. The equations of motion for the relative position and orientation, the terminal condition, the state and control constraints and the cost functional are presented. Afterwards a direct approach is applied to solve the optimal control problem. In the fourth part the numerical results for a flyaround maneuver with a stable spinning target are presented and discussed. Thereafter a short introduction in simulating the trajectories in a virtual reality is given. An outlook of possible extensions concludes the paper.

2. SETUP OF THE RENDEZVOUS MODEL

The motion of each spacecraft consists of two independent 3D subsystems, the relative position and the relative orientation. The equations of motion for the relative position

are the so called Clohessy-Wilshire equations. Further information on orbit analysis and design can be found in Schilling (2011). Note that throughout this work we shorten notation by omitting the time dependency if no specific time instant is considered.

2.1 Relative position

Assuming circular orbits the subsystem of the relative position is derived via Kepler's equation for the two body problem and a linear Taylor expansion, see, e.g., Alfriend et al. (2009). As a result, a second order ordinary differential equation system for the position of a body relative to a moving reference point in orbit is obtained. Throughout this work, we consider the uncontrolled target to be that reference point. Apart from the standard state variables x , y and z denoting the relative distance in radial, direction of movement and out of orbit component respectively, we introduce the additional state variables v_x , v_y and v_z denoting the respective velocities to transform the system into six first order differential equations.

$$\dot{x} = v_x \quad (1)$$

$$\dot{y} = v_y \quad (2)$$

$$\dot{z} = v_z \quad (3)$$

$$\dot{v}_x = 2nv_y + 3n^2x + \frac{u_x}{m} \quad (4)$$

$$\dot{v}_y = -2nv_x + \frac{u_y}{m} \quad (5)$$

$$\dot{v}_z = -n^2z + \frac{u_z}{m} \quad (6)$$

Here m denotes the mass of the servicer spacecraft and n is the mean motion of the reference point, i.e. the angular velocity in orbit. For the dynamics it can be seen that the z coordinate is decoupled from the x - y -system, i.e. the out of orbit motion is independent of the motion in the orbit plane. If no controls u_x , u_y , u_z are applied an analytical solution is known, again see Alfriend et al. (2009). Yet for non vanishing controls no analytical solution is known. For this reason, we evaluate the dynamics numerically.

2.2 Orientation

In space mission modelling one commonly uses the quaternion representation to derive the equations for the orientation of a satellite. Quaternions are a special parametrization of the Euler-axis/angle description which is nonsingular and continuous. A quaternion can be considered as a four dimensional vector $\mathbf{q} := [q_1, q_2, q_3, q_4]^\top$. The first three components are called the vector part and the fourth is the scalar component. The denomination results from the fact that a quaternion represents a rotation around an axis in three dimensional space, i.e. the vector part, by a specific angle given by the scalar component. To get a unique description of the orientation, the quaternions are supposed to be of length one, so called unit quaternions. Further information on quaternions and their relation to other rotation types can be found in Tewari (2007).

The first derivative of quaternions is given by

$$\dot{\mathbf{q}} = \frac{1}{2} \begin{bmatrix} \boldsymbol{\omega} \\ 0 \end{bmatrix} \otimes \mathbf{q},$$

where $\boldsymbol{\omega} := [\omega_x, \omega_y, \omega_z]^\top$ is the vector of the angular velocities and \otimes represents the quaternion multiplication,

see also Stevens and Lewis (2003) for a derivation. In its components the dynamics of the quaternions are of the form

$$\begin{bmatrix} \dot{q}_1^\alpha \\ \dot{q}_2^\alpha \\ \dot{q}_3^\alpha \\ \dot{q}_4^\alpha \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & \omega_z^\alpha & -\omega_y^\alpha & \omega_x^\alpha \\ -\omega_z^\alpha & 0 & \omega_x^\alpha & \omega_y^\alpha \\ \omega_y^\alpha & -\omega_x^\alpha & 0 & \omega_z^\alpha \\ -\omega_x^\alpha & -\omega_y^\alpha & -\omega_z^\alpha & 0 \end{bmatrix} \begin{bmatrix} q_1^\alpha \\ q_2^\alpha \\ q_3^\alpha \\ q_4^\alpha \end{bmatrix} \quad \alpha \in \{S, T\}, \quad (7)$$

where S and T denote the service and target spacecraft respectively.

Note that the angular velocity of a tumbling object is typically not constant with respect to time. Denoting the inertia tensor with \mathbf{J} we incorporate that aspect using Euler's gyroscopic equation

$$\dot{\mathbf{J}} \cdot \boldsymbol{\omega} + \mathbf{J} \cdot \dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times (\mathbf{J} \cdot \boldsymbol{\omega}) = \mathbf{m},$$

which provides the needed information, see, e.g., Chobotov (1991). Under the assumptions that mass distribution is constant over time, i.e. $\dot{\mathbf{J}} = 0$, and the regarded system coincides with the principal axis of the body, that is $J_{ik} = 0$ for all $i \neq k$, we obtain the set of first order equations

$$\dot{\omega}_x^S = \frac{1}{J_{xx}^S} (\omega_y^S \omega_z^S (J_{yy}^S - J_{zz}^S) + m_x) \quad (8)$$

$$\dot{\omega}_y^S = \frac{1}{J_{yy}^S} (\omega_x^S \omega_z^S (J_{zz}^S - J_{xx}^S) + m_y) \quad (9)$$

$$\dot{\omega}_z^S = \frac{1}{J_{zz}^S} (\omega_x^S \omega_y^S (J_{xx}^S - J_{yy}^S) + m_z) \quad (10)$$

$$\dot{\omega}_x^T = \frac{1}{J_{xx}^T} (\omega_y^T \omega_z^T (J_{yy}^T - J_{zz}^T)) \quad (11)$$

$$\dot{\omega}_y^T = \frac{1}{J_{yy}^T} (\omega_x^T \omega_z^T (J_{zz}^T - J_{xx}^T)) \quad (12)$$

$$\dot{\omega}_z^T = \frac{1}{J_{zz}^T} (\omega_x^T \omega_y^T (J_{xx}^T - J_{yy}^T)). \quad (13)$$

where $\mathbf{m} := [m_x, m_y, m_z]^\top$ defines the momentum vector. Note that no momentum \mathbf{m} is applied on the right hand side of the target satellite as it is supposed to be uncontrolled.

With these equations of motion we obtain a system of twenty first order ordinary differential equations (1)–(13). Here, we denote the combined state vector by

$$\mathbf{x} := [x, y, z, v_x, v_y, v_z, \omega_x^S, \omega_y^S, \omega_z^S, \omega_x^T, \omega_y^T, \omega_z^T, q_1^S, q_2^S, q_3^S, q_4^S, q_1^T, q_2^T, q_3^T, q_4^T]^\top$$

and the control vector by

$$\mathbf{u} := [u_x, u_y, u_z, m_x, m_y, m_z]^\top.$$

2.3 Terminal condition

To model the docking of the two spacecraft some terminal constraints have to be introduced. We define a docking point for each satellite in their body fixed coordinate system, $\mathbf{d}^S := [d_1^S, d_2^S, d_3^S]^\top$ and $\mathbf{d}^T := [d_1^T, d_2^T, d_3^T]^\top$, whose local positions and velocities have to match at the end of the maneuver. Similar conditions can be found in, e.g., Alfriend et al. (2009) and Boyarko (2010). To get a common representation, all components are converted to the earth centered inertial coordinate system. We denote the corresponding transformation matrix from the rotated body fixed coordinates to the inertial system by R^\top and add the index \mathcal{E} to variables defined in the earth central inertial coordinate system. Computing the transformation

matrix from the components of the related quaternion, see, e.g., Tewari (2007), we obtain

$$R = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 + q_3q_4) & 2(q_1q_3 - q_2q_4) \\ 2(q_1q_2 - q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2q_3 + q_1q_4) \\ 2(q_1q_3 + q_2q_4) & 2(q_2q_3 - q_1q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \quad (14)$$

Multiplication with this matrix performs the transformation of a vector from the unrotated coordinate system to the rotated one. The inverse transformation, as we need it in this case, can be done by multiplication with the transposed matrix of R . Applying the transformation for both the service and the target satellites, the local difference of the docking point is given by

$$\mathbf{d}_{\mathcal{E}}^S - \mathbf{d}_{\mathcal{E}}^T = \mathbf{r} + R^{S\top} \mathbf{d}^S - R^{T\top} \mathbf{d}^T, \quad (15)$$

with $\mathbf{r} := [x, y, z]^\top$ being the cartesian distance of the two spacecrafts. Similarly, the angular velocity of each satellite in the earth centered inertial frame is given by

$$\boldsymbol{\omega}_{\mathcal{E}}^\alpha = R^{\alpha\top} \boldsymbol{\omega}^\alpha - \begin{bmatrix} 0 \\ 0 \\ n \end{bmatrix} \quad \alpha \in \{S, T\}.$$

Hence, we obtain the difference of the velocity via

$$\dot{\mathbf{d}}_{\mathcal{E}}^S - \dot{\mathbf{d}}_{\mathcal{E}}^T = \dot{\mathbf{r}} + \boldsymbol{\omega}_{\mathcal{E}}^S \times (R^{S\top} \mathbf{d}^S) - \boldsymbol{\omega}_{\mathcal{E}}^T \times (R^{T\top} \mathbf{d}^T). \quad (16)$$

To simplify this equation we assume that the angular velocity and the orientation of both spacecrafts coincide at the end of the time interval, i.e.,

$$\mathbf{q}^T(t_f) - \mathbf{q}^S(t_f) = \mathbf{0} \quad (17)$$

$$\boldsymbol{\omega}^T(t_f) - \boldsymbol{\omega}^S(t_f) = \mathbf{0}. \quad (18)$$

Note that the docking point definition is required to be conform with these assumptions. To illustrate this point, consider the docking points to be located at the back of the satellites. Then, since the satellites would have to overlap, there exists no physically possible configuration where these points and the satellite orientations coincide. Assuming (17) and (18) to hold we obtain

$$R^T(t_f) = R^S(t_f) = R(t_f)$$

$$\boldsymbol{\omega}_{\mathcal{E}}^T(t_f) = \boldsymbol{\omega}_{\mathcal{E}}^S(t_f) = \boldsymbol{\omega}_{\mathcal{E}}(t_f)$$

and the docking conditions (15) and (16) simplify to

$$\begin{aligned} R(t_f)^\top (\mathbf{d}^T - \mathbf{d}^S) - \mathbf{r} &= \mathbf{0} \\ \boldsymbol{\omega}_{\mathcal{E}}(t_f) \times R(t_f)^\top (\mathbf{d}^T - \mathbf{d}^S) - \dot{\mathbf{r}} &= \mathbf{0}. \end{aligned}$$

2.4 State and control constraints

As pointed out in the previous subsection, the case of colliding satellites has to be treated. To exclude such an occurrence, we first introduce a spherical safety area around each satellite with radius $r^S > 0$ and $r^T > 0$ respectively. In our case, the usage of spheres is reasonable as the geometry of the satellites is supposed to be simple. Based on the safety areas, we add the state constraint

$$x^2 + y^2 + z^2 \geq (r^S + r^T)^2$$

to the optimal control problem which resolves the possible collision issue. Note that the safety area needs to be chosen carefully since the state constraint together with the terminal condition may exclude the existence of a solution.

Apart from the states we also consider the controls to be constrained. In particular, we suppose that the thrust vector $\mathbf{u} := [u_x, u_y, u_z]^\top$ is bounded via

$$u_x^2 + u_y^2 + u_z^2 \leq u_{\max}$$

and each momentum is bounded via

$$|m_i| \leq m_{\max} \quad i = x, y, z.$$

The choice of the limitations in the momentum control is motivated by the fact that the satellites attitude control system can be realized via control moment gyroscopes. These can cause a maximal momentum m_{\max} around each axis. The bound of the thrust vector is due to future extensions of the model where only one thruster is installed and the spacecraft has to rotate to accelerate in a specific direction.

As the thrusters are mounted on the satellite their direction depends on the current attitude of the spacecraft. Therefore we transform the calculated thrust u_x , u_y and u_z from earth centered coordinates into the body fixed system. The transformation is done by multiplication with the rotation matrix R defined in (14) and reveals

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} := R \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix}$$

where u_1 , u_2 and u_3 denote the controls in the body fixed coordinates. Note that as R is orthogonal the length of \mathbf{u} remains unchanged by this transformation, and hence the upper bound u_{\max} on the controls in the body fixed and the earth centered coordinates are identical.

2.5 Cost functional

Throughout this work we consider *minimizing* the following Bolza type cost functional over the set of allowed control functions $\hat{\mathbf{u}}(t) := [\mathbf{u}(t), \mathbf{m}(t)]^\top$ and terminal times t_f . We define the cost functional as follows

$$J(\hat{\mathbf{u}}(t), t_f) = l_{t_f} t_f + \int_0^{t_f} l_u \|\mathbf{u}(t)\|_2^2 + l_m \|\mathbf{m}(t)\|_2^2 dt. \quad (19)$$

This functional represents a combination of the terminal time, the position controls and the orientation controls. The non-negative constants l_{t_f} , l_u and l_m can be used to force scenarios such as time-optimality or minimal control effort. Note that the 2-norm $\|\cdot\|_2$ has to be used if results shall hold for both controls $(u_1, u_2, u_3)^\top$ and $(u_x, u_y, u_z)^\top$.

3. DISCRETIZATION AND IMPLEMENTATION

To solve the optimal control problem

$$\text{Minimize } J(\hat{\mathbf{u}}(t), t_f)$$

subject to the dynamics (1) – (13)

with initial and terminal conditions

$$\mathbf{x}(0) = \mathbf{x}_0$$

$$\mathbf{0} = \mathbf{q}^T(t_f) - \mathbf{q}^S(t_f)$$

$$\mathbf{0} = \boldsymbol{\omega}^T(t_f) - \boldsymbol{\omega}^S(t_f)$$

$$\mathbf{0} = R(t_f)^\top (\mathbf{d}^T - \mathbf{d}^S) - \mathbf{x}$$

$$\mathbf{0} = \boldsymbol{\omega}_{\mathcal{E}}(t_f) \times R(t_f)^\top (\mathbf{d}^T - \mathbf{d}^S) - \dot{\mathbf{x}} \quad (20)$$

and constraints

$$u_{\max} \geq u_x^2(t) + u_y^2(t) + u_z^2(t) \quad \forall t \in [0, t_f]$$

$$m_{\max} \geq |m_i(t)| \quad i = x, y, z \quad \forall t \in [0, t_f]$$

$$x^2(t) + y^2(t) + z^2(t) \geq (r^S + r^T)^2 \quad \forall t \in [0, t_f] \quad (21)$$

a direct approach can be applied. Using a full discretization with fixed step size $\Delta t = t_f/N$ where N is the number of steps, the discretized dynamics of the state \mathbf{x} and the discretized constraints (20), (21) form the constraints of the resulting finite nonlinear program. The optimization variable now contains not only the discretization of the control $\hat{\mathbf{u}}$ and the free terminal time t_f , but also the discretization of the entire state trajectory. With the exponent k we denote the state of a variable at the k th discretization point, i.e. $x^k = \mathbf{x}(k \cdot \Delta t)$.

Since the gyroscopic equation is stiff the implicit trapezoidal method

$$\frac{\mathbf{x}^{k+1} - \mathbf{x}^k}{\Delta t} = \frac{1}{2}(\mathbf{f}(\mathbf{x}^k, \mathbf{u}^k, t^k) + \mathbf{f}(\mathbf{x}^{k+1}, \mathbf{u}^{k+1}, t^{k+1}))$$

$$k = 0, 1, \dots, N-1$$

is used to discretize the dynamics of the state. Note that the components of the angular velocity would soar even for very small step-sizes if, e.g., Euler's method was used. We implemented the discretized optimal control problem in the modelling language AMPL and solved it using the interior point optimizer IpOpt. For details on AMPL and IpOpt see, e.g., Fourer et al. (2003) and Wächter and Biegler (2006). For the discretized problem, the constraints are defined pointwise at the discretization points. To avoid constraint violations at intermediate time instances, adaptation techniques for the discretization grid may be used, see, e.g., Betts and Huffman (1998), or the bounds may be tightened.

4. NUMERICAL RESULTS

To illustrate our results we consider a flyaround maneuver, that is a situation where the service satellite is located on the wrong side of the target pointing with its docking interface towards it. Here, we assume that the docking interface of the servicer is located on the 'front' whereas the docking point of the target is at its 'back', e.g. the targets thruster where some kind of hook clasps. The radius of the safety area for each satellite is chosen to be of size one. For both satellites we assume that they are axially symmetric with respect to the y axis which is justified for the regarded satellite geometries. Note that the choice of the docking points enables us to use the simplified equations for the terminal condition (17) and (18) and creates no impossible final states.

Upon start of the maneuver, we suppose that the servicer is in a non-rotated initial state and its angular velocity is zero. The target also starts in a non-rotating initial state but in contrast to the servicer it is rotating with a constant angular velocity of $3^\circ/\text{s}$ around its y axis resulting in a stable motion with respect to time. The initial values are as follows:

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} v_{x0} \\ v_{y0} \\ v_{z0} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},$$

$$\mathbf{q}^S = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \boldsymbol{\omega}^S = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},$$

$$\mathbf{q}^T = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \boldsymbol{\omega}^T = \begin{bmatrix} 0 \cdot 0.017453 \\ 3 \cdot 0.017453 \\ 0 \cdot 0.017453 \end{bmatrix}.$$

Geometrically, the above conditions force the servicer to fulfill three tasks: It has to fly around the target, turn around to align the docking points and adopt the rotation such that the angular velocities coincide.

Within the cost functional (19) we consider the weights $l_{t_f} = 1$, $l_u = 1$ and $l_m = 1$, i.e., the terminal time and the control costs are weighted equally. The remaining constants of the optimal control problem are displayed in Table 1.

variable	value	description
a	7071000	orbit radius [m]
GM	$398 \cdot 10^{12}$	gravitational constant [$N(m/kg)^2$]
n	$\sqrt{\frac{GM}{a^3}}$	mean motion [1/s]
m	200	satellite mass [kg]
u_{\max}	0.15	maximum thrust [N]
m_{\max}	1	maximum torque [Nm]
J_{xx}^T	1000	Targ. angular mass around x [kg/m^2]
J_{yy}^T	2000	Targ. angular mass around y [kg/m^2]
J_{zz}^T	1000	Targ. angular mass around z [kg/m^2]
J_{xx}^S	2000	Serv. angular mass around x [kg/m^2]
J_{yy}^S	5000	Serv. angular mass around y [kg/m^2]
J_{zz}^S	2000	Serv. angular mass around z [kg/m^2]
r_{target}^S	1	safety-area around target [m]
r_{servicer}^S	1	safety-area around servicer [m]
d^T	$[0 \ -1 \ 0]^T$	docking point target [m]
d^S	$[0 \ 1 \ 0]^T$	docking point servicer [m]

Table 1. Constants of optimal control problem

To convert the optimal control problem into a finite nonlinear program we used a discretization with $N = 370$ steps. The results of our calculations are displayed in Figures 1 and 2 which show the states and controls over time. For the optimal terminal time we obtain $t_f = 369.61$ seconds, which is about 6.16 minutes.

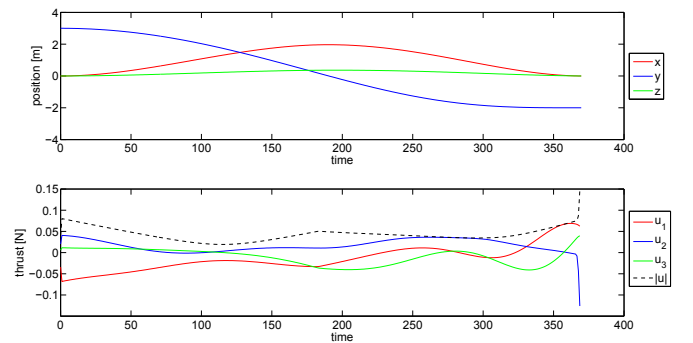


Fig. 1. Position and thruster control of the servicer

From Figure 1 one observes that the position control never reaches the maximum. The fact that the allowed thrust is never applied is due to the chosen weights in the cost functional. The controls are smooth, except one point at about 190 seconds, but there occurs no bang-bang behaviour. Because of the state constraints the satellite cannot fly to its terminal position in a straight line. In the beginning it accelerates in positive x direction – which is

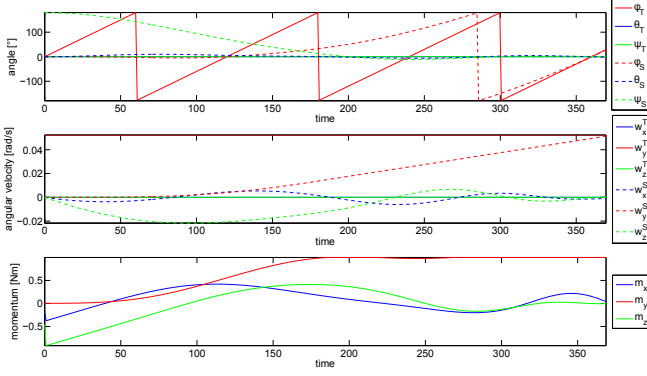


Fig. 2. Orientation, angular velocity and momentum controls of target and servicer (dashed)

due to the initial orientation – performed by a negative control of u_1 . Note that the applied control has to be interpreted with respect to the current attitude of the satellite, cf. Section 2.4. Upon termination of the maneuver the negative control in y direction represents the breaking to prevent collision with the target and to obtain zero relative velocity. As expected, the position in x and y nearly produce a symmetrical arc with respect to half the maneuver time.

The orientations in Figure 2 are plotted in Euler- yxz -angles with the angle notation ϕ , θ and ψ . One observes that the spin of the target around the y -axis is periodical. In the first part of the maneuver the main momentum control of the servicer is applied to perform the turning. To this end, the almost maximal momentum is applied around the z -axis and some additional correction in x direction. This control first accelerates the angular velocity of the servicer around the turning axis and then slows it down such that upon termination of the maneuver the angular velocity around the x - and z -axis are zero. After about 50 seconds the momentum around the y -axis is increasing. It reaches the maximal possible amount and nearly keeps this value until the end of the maneuver. This is the control for adapting the angular velocity around y . The acceleration of ω_y is timed such that upon termination both the angular velocity and the orientation of servicer and target are equal.

Computing the total control costs of the thrust and the momentum control, we obtain

$$u_{\text{total}} = \Delta t \sum_{k=0}^{N-1} (u_x^k)^2 + (u_y^k)^2 + (u_z^k)^2 = 15.7662$$

$$m_{\text{total}} = \Delta t \sum_{k=0}^{N-1} (m_x^k)^2 + (m_y^k)^2 + (m_z^k)^2 = 295.5767.$$

Hence, the costs of the momentum control outweigh the costs of the position control for this particular example. The optimal value of the cost functional sums up to

$$J = t_f + u_{\text{total}} + m_{\text{total}} = 680.9548.$$

To demonstrate that the calculated control depends on the state constraints we calculated the optimal trajectory with the same initial values omitting this constraint. In Figure 3 one can see that with the safety areas the servicer flies a nearly circular arc around the target. Without the

constraint the flight path is much more direct and the servicer would crash into the target.

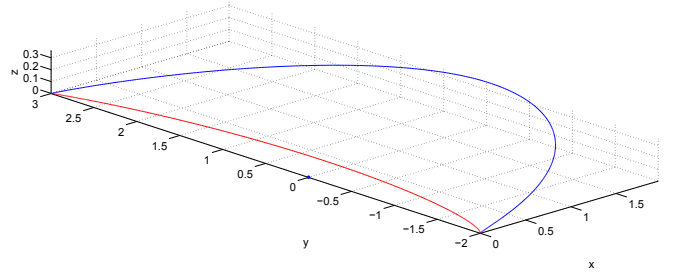


Fig. 3. Trajectory of the servicer with state constraint (blue) and without state constraint (red)

5. SIMULATION

Since the computed results are hard to imagine from the plots, we developed a virtual reality where the calculated data can be imported and rendered images of the maneuver can be created. These images are then combined to a video showing the entire rendezvous maneuver. The virtual reality was created with the open source software Blender 3D. Using an additional import tool the trajectory data is bound to the corresponding satellite. Moreover, the calculated quaternions are used directly to rotate the models. For information on Blender see, e.g., van Gumster (2009). The satellite models were designed based on the first design studies of the german aerospace center, see Sellmaier et al. (2010). In Figure 4 one can see an image of the two satellites in the created virtual reality.

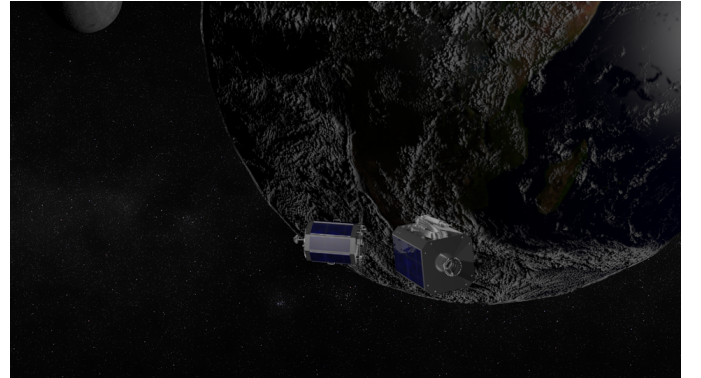


Fig. 4. Rendered picture of the satellites

6. OUTLOOK

The shown example only covers a stable rotating target satellite, i.e. all angular velocity components are constant with respect to time. Yet, the presented model can also be used to calculate optimal solutions for a tumbling target which may occur if a satellite runs out of fuel or energy to keep its orientation. This case is under investigation at the moment to handle the additional difficulty of a moving target point. Note that it is possible to calculate the maximal tumbling motion which still allows a docking maneuver to be performed. If this maximum is exceeded, the service satellite can be fitted with an additional manipulator arm to grasp the target to establish a connection.

To optimize such a so called berthing maneuver the model derived in this work has to be modified by adding a model of a robot arm, see also Fehse (2003) for information on the differences between docking and berthing. For one, future research will concern computing optimal berthing maneuvers, but also to derive low cost feedback controllers for both docking and berthing situations.

ACKNOWLEDGEMENTS

We thank Prof. Dr. K. Schilling, University of Würzburg, for suggesting this work and for fruitful discussions.

REFERENCES

- Alfriend, K., Gurfil, P., and Vadali, S. (2009). *Spacecraft Formation Flying: Dynamics, Control, and Navigation*. Elsevier Astrodynamics Series. Elsevier Science & Technology.
- Betts, J. and Huffman, W.P. (1998). Mesh refinement in direct transcription methods for optimal control. *Optimal Control Applications and Methods*, 19, 1–21. doi:10.1002/(SICI)1099-1514(199801/02)19:1(1::AID-OCA616)3.0.CO;2-Q.
- Boyarko, G. (2010). *Spacecraft guidance strategies for proximity maneuvering and close approach with a tumbling object*. Ph.D. thesis, Naval Postgraduate School.
- Chobotov, V. (1991). *Spacecraft attitude dynamics and control*. Orbit, a foundation series. Krieger.
- Fehse, W. (2003). *Automated rendezvous and docking of spacecraft*. Cambridge aerospace series. Cambridge University Press.
- Fourer, R., Gay, D., and Kernighan, B. (2003). *AMPL: a modeling language for mathematical programming*. Thomson/Brooks/Cole.
- Klinkrad, H. (2011). The re-entry of the rosat satellite and the risks. URL http://www.dlr.de/dlr/en/Portaldata/1/Resources/documents/ROSAT_klinkrad_en.pdf.
- NASA Orbital Space Debris Program (2011). Orbital Debris Frequently Asked Questions. URL <http://www.orbitaldebris.jsc.nasa.gov/faqs.html>.
- Pickup, A. (2009). Spacewatch. URL <http://www.guardian.co.uk/science/2009/feb/18/spacewatch-satellite-collision>.
- Schilling, K. (2011). Mission analysis for low-earth-observation missions with spacecraft formations. In *NATO RTO Lecture Series – Small Satellite Formations for Distributed Surveillance: System Design and Optimal Control Considerations*. RTO-EN-SCI-231. NATO RTO Lecture Series, Würzburg, Germany.
- Sellmaier, F., Boge, T., Spurmann, J., Gully, S., Rupp, T., and Huber, F. (2010). On-orbit servicing missions: Challenges and solutions for spacecraft operations. In *SpaceOps 2010 Conference*. German Space Operations Center (GSOC), Huntsville, Alabama, USA.
- Stevens, B.L. and Lewis, F.L. (2003). *Aircraft Control and Simulation*. Wiley-Interscience.
- Tewari, A. (2007). *Atmospheric and Space Flight Dynamics: Modeling and Simulation with MATLAB and Simulink*. Birkhäuser Basel, 1st edition.
- van Gumster, J. (2009). *Blender For Dummies*. For Dummies.
- Wächter, A. and Biegler, L. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106, 25–57. doi:10.1007/s10107-004-0559-y.